

Amendments to the Claims:

This listing of claims replaces all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A computerized method of managing resource usage by one or more resource entities, wherein each resource entity is configured to represent an entity that consumes one or more resources, the method comprising:

in an object-based computing system, continuously determining by a resource manager with which each resource entity is registered whether a resource has reached a critical level;

when it is determined that a resource has reached a critical level, selecting by the resource manager, a first resource entity based on one or more criteria, wherein the one or more criteria include a priority value for each resource entity and wherein the selected resource entity has a relatively lower priority than other one or more resource entities that are not selected; and

when it is determined that a resource has reached a critical level, requesting by the resource manager that the selected resource entity **[[to]]** change its resource consumption state to a more restrictive state by performing the request through a list to which a plurality of resource entities are registered.

2. (Canceled)

3. (Original) A method as recited in claim 1, wherein the one or more criteria include a current resource consumption state of each resource entity and wherein the selected resource entity has a relatively less restrictive current resource consumption state than one or more other resource entities that are not selected.

4. (Previously Presented) A method as recited in claim 1, further comprising:
- when the selected resource entity does not change its resource consumption state when requested to do so, determining by the resource manager whether to request the selected resource entity a second time to change its resource consumption state to the previously requested more restrictive state; and
- when it is determined to request the selected resource entity a second time, requesting by the resource manager that the selected resource entity a second time ~~to~~ change its resource consumption state to the previously requested more restrictive state.
5. (Previously Presented) A method as recited in claim 4, further comprising terminating by the resource manager the selected resource entity when it is determined not to request the selected resource entity a second time.
6. (Original) A method as recited in claim 5, wherein determining whether to request the selected resource entity a second time is based on whether the selected resource entity has failed to change its resource consumption when requested to do so within a predetermined time limit.
7. (Original) A method as recited in claim 5, wherein determining whether to request the selected resource entity a second time is based on whether a resource is still at a critical level.
8. (Previously Presented) A method as recited in claim 5, wherein determining whether to request the selected resource entity a second time is based on whether a second resource entity is operable to be selected and requested to change its resource consumption to a more restrictive state.

9. (Original) A method as recited in claim 5, wherein determining whether to request the selected resource entity a second time is based on whether the selected resource entity has indicated a reduction in resource usage.
10. (Previously Presented) A method as recited in claim 1, wherein the selected resource entity is operable to have or be requested to have a resource consumption state selected from a group consisting of a NORMAL state, a FAVOR_SPACE state, an ESSENTIAL_ONLY state, a TERMINATE_GRACEFULLY state, and a TERMINATE_ABRUPTLY state.
11. (Original) A method as recited in claim 1, wherein the resource is selected from a group consisting of a Java heap, a plurality of file descriptors, video RAM, a native heap, a hardware resource, and persistent storage.
12. (Original) A method as recited in claim 1, wherein the resource entity is contained within an entity selected from a group consisting of an application, an applet, and an Xlet.
13. (Previously Presented) A method as recited in claim 1, wherein the list is a ResourceNotificationList.
14. (Original) A method as recited in claim 13, further comprising obtaining the selected resource entity's current resource consumption state through the ResourceNotificationList prior to requesting the selected resource entity to change its resource consumption state to a more restrictive state.
15. (Previously Presented) A method as recited in claim 1, further comprising:

when the selected resource entity changes to the requested state, determining again by the resource manager whether any resource is still at a critical level;

when it is determined that a resource is at a critical level, selecting by the resource manager a second resource entity based on one or more criteria; and

when it is determined that a resource is at a critical level, requesting by the resource manager that the second selected resource entity to change its resource consumption state to a more restrictive state.

16. (Original) A method as recited in claim 15, wherein the first selected resource entity equals the second selected resource entity.

17. (Previously Presented) A method as recited in claim 16, further comprising telling the second selected resource entity by the resource manager to return to its least restrictive resource consumption state when it is determined that no resources have reached a critical level.

18. (Original) A method as recited in claim 1, further comprising forwarding the request to a second resource entity whose resource usage is managed by an application associated with the first resource entity.

19. (Previously Presented) A computer readable medium containing computer code having instructions for managing resource usage by one or more resource entities, wherein each resource entity is configured to represent an entity that consumes one or more resources, the computer readable medium comprising:

computer code for continuously determining by the resource manager with which each resource entity is registered whether a resource in an object-based computing system has reached a critical level;

computer code for selecting by the resource manager, a first resource entity based on one or more criteria when it is determined that a resource has reached a critical level, wherein the one or more criteria include a priority value for each resource entity and wherein the selected resource entity has a relatively lower priority than other one or more resource entities that are not selected; and

computer code for requesting by the resource manager that the selected resource entity change its resource consumption state to a more restrictive state when it is determined that a resource has reached a critical level by performing the request through a list to which a plurality of resource entities are registered.

20. (Canceled)

21. (Original) A computer readable medium as recited in claim 19, wherein the one or more criteria include a current resource consumption state of each resource entity and wherein the selected resource entity has a relatively less restrictive current resource consumption state than one or more other resource entities that are not selected.

22. (Previously Presented) A computer readable medium as recited in claim 19, further comprising:

computer code for determining by the resource manager whether to request the selected resource entity a second time to change its resource consumption state to the previously

requested more restrictive state when the selected resource entity does not change its resource consumption state when requested to do so; and

computer code for requesting by the resource manager that the selected resource entity a second time to change its resource consumption state to the previously requested more restrictive state when it is determined to request the selected resource entity a second time.

23. (Previously Presented) A computer readable medium as recited in claim 22, further comprising computer code for terminating by the resource manager the selected resource entity when it is determined not to request the selected resource entity a second time.

24. (Original) A computer readable medium as recited in claim 23, wherein determining whether to request the selected resource entity a second time is based on whether the selected resource entity has failed to change its resource consumption when requested to do so within a predetermined time limit.

25. (Original) A computer readable medium as recited in claim 23, wherein determining whether to request the selected resource entity a second time is based on whether a resource is still at a critical level.

26. (Previously Presented) A computer readable medium as recited in claim 23, wherein determining whether to request the selected resource entity a second time is based on whether a second resource entity is operable to be selected and requested to change its resource consumption to a more restrictive state.

27. (Original) A computer readable medium as recited in claim 23, wherein determining whether to request the selected resource entity a second time is based on whether the selected resource entity has indicated a reduction in resource usage.

28. (Previously Presented) A computer readable medium as recited in claim 19, wherein the selected resource entity is operable to have or be requested to have a resource consumption state selected from a group consisting of a NORMAL state, a FAVOR_SPACE state, an ESSENTIAL_ONLY state, a TERMINATE_GRACEFULLY state, and a TERMINATE_ABRUPTLY state.

29. (Original) A computer readable medium as recited in claim 19, wherein the resource is selected from a group consisting of a Java heap, a plurality of file descriptors, video RAM, and a native heap.

30. (Original) A computer readable medium as recited in claim 19, wherein the resource entity is contained within an entity selected from a group consisting of an application, an applet, and an Xlet.

31. (Currently Amended) A ~~method~~ computer readable medium as recited in claim 18, wherein the list is a ResourceNotificationList.

32 (Original) A computer readable medium as recited in claim 31, further comprising computer code for obtaining the selected resource entity's current resource consumption state through the ResourceNotificationList prior to requesting the selected resource entity to change its resource consumption state to a more restrictive state.

33. (Previously Presented) A computer readable medium as recited in claim 19, further comprising:

computer code for determining again by the resource manager whether any resource is still at a critical level when the selected resource entity changes to the requested state;

computer code for selecting by the resource manager a second resource entity based on one or more criteria when it is determined that a resource is at a critical level; and

computer code for requesting by the resource manager that the second selected resource entity change its resource consumption state to a more restrictive state when it is determined that a resource is at a critical level.

34. (Original) A computer readable medium as recited in claim 33, wherein the first selected resource entity equals the second selected resource entity.

35. (Previously Presented) A computer readable medium as recited in claim 34, further comprising computer code for telling the second selected resource entity by the resource manager to return to its least restrictive resource consumption state when it is determined that no resources have reached a critical level.

36. (Original) A computer readable medium as recited in claim 19, further comprising computer code for forwarding the request to a second resource entity whose resource usage is managed by an application associated with the first resource entity.

37. (Currently Amended) **[[A]]** An object-based computer system operable to manage resource usage by one or more resource entities, wherein each resource entity is configured to

represent an entity that consumes one or more resources, comprising:

a memory; and

a processor coupled to the memory, wherein at least one of the memory and the processor are adapted to provide:

select a first resource entity based on one or more criteria when it is determined that a resource has reached a critical level, wherein the one or more criteria include a priority value for each resource entity and wherein the selected resource entity has a relatively lower priority than other one or more resource entities that are not selected; and

request the selected resource entity to change its resource consumption state to a more restrictive state when it is determined that a resource has reached a critical level by performing the request through a list to which a plurality of resource entities are registered.

38. (Canceled)

39. (Original) A computer system as recited in claim 37, wherein the one or more criteria include a current resource consumption state of each resource entity and wherein the selected resource entity has a relatively less restrictive current resource consumption state than one or more other resource entities that are not selected.

40. (Previously Presented) A computer system as recited in claim 37, further comprising:
when the selected resource entity does not change its resource consumption state when requested to do so, determining by the resource manager whether to request the selected resource entity a second time to change its resource consumption state to the previously requested more restrictive state; and

when it is determined to request the selected resource entity a second time, requesting by the resource manager that the selected resource entity a second time ~~to~~ change its resource consumption state to the previously requested more restrictive state.

41. (Original) A computer system as recited in claim 40, wherein when it is determined not to request the selected resource entity a second time, terminating the selected resource entity.

42. (Original) A computer system as recited in claim 41, wherein determining whether to request the selected resource entity a second time is based on whether the selected resource entity has failed to change its resource consumption when requested to do so within a predetermined time limit.

43. (Original) A computer system as recited in claim 41, wherein determining whether to request the selected resource entity a second time is based on whether a resource is still at a critical level.

44. (Previously Presented) A computer system as recited in claim 41, wherein determining whether to request the selected resource entity a second time is based on whether a second resource entity is operable to be selected and requested to change its resource consumption to a more restrictive state.

45. (Original) A computer system as recited in claim 41, wherein determining whether to request the selected resource entity a second time is based on whether the selected resource entity has indicated a reduction in resource usage.

46. (Previously Presented) A computer system as recited in claim 37, further comprising:
when the selected resource entity changes to the requested state, determining again by the resource manager whether any resource is still at a critical level;
when it is determined that a resource is at a critical level, selecting by the resource manager a second resource entity based on one or more criteria; and
when it is determined that a resource is at a critical level, requesting by the resource manager that the second selected resource entity change its resource consumption state to a more restrictive state.

47. (Original) A computer system as recited in claim 46, wherein the first selected resource entity equals the second selected resource entity.

48. (Previously Presented) A computer system as recited in claim 47, further comprising telling the second selected resource entity by the resource manager to return to its least restrictive resource consumption state when it is determined that no resources have reached a critical level.

49. (Original) A computer system as recited in claim 37, further comprising forwarding the request to a second resource entity whose resource usage is managed by an application associated with the first resource entity.